Heuristic Motion Planning in Dynamic Environments using Velocity Obstacles

Paolo Fiorini[†,‡]

Zvi Shiller[‡]

[†] Jet Propulsion laboratory
California Institute of Technology
Pasadena, California 91109

[‡] Department of Mechanical, Aerospace
and Nuclear Engineering
University of California, Los Angeles
Los Angeles, California 90024

Abstract

This paper presents heuristic methods for motion planning in dynamic environments, based on the concept of Velocity Obstacle (VO). Representing the moving obstacles by their VO defines the set of safe velocities for all avoidance maneuvers. Selecting any velocity in that set allows to generate trajectories that ensure that the moving robot dots not collide with the static and moving obstacles, reach the goal, and possibly minimize motion time. In this paper we demonstrate two heuristic strategies: 1. selecting the maximum velocity along the line to the target, and 2. selecting the maximum feasible velocity within a specified angle from the straight line to the target. Examples are presented that demonstrate the use of these heuristics for planning the motions of an intelligent vehicle moving from the fast lane to the exit ramp. The heuristic trajectories are compared to the trajectories computed with a global search.

1 Introduction

We consider the problem of motion planning for a robot (mobile robot or intelligent vehicle) moving in dynamic environments, populated with stationary and moving obstacles. The objectives of the motion plans are to avoid collision with the stationary and the moving obstacles, reach a desired goal, and minimize motion time, while respecting robot's dynamics. Examples of such environments include: multiple-robot assembly, assembly of parts moving on conveyor belts, intelligent vehicles traveling on smart highways, and airspace surrounding airports.

Motion planning in dynamic environments is significantly different from, and considerably more difficult than, the widely studied static problem. Motion planning in static environments can be guaranteed to find a solution that meets the desired goals if one exists, whereas motion planning in dynamic environments is intractable [1 1]. In other words, it is not possible to guarantee the survival of the robot, let alone ensure that it reaches its destination. The objectives of motion planning in truly dynamic environments are therefore first to survive, and then to accomplish as many of the other goals, such as reaching the destination and minimizing motion time, as possible.

Motion planning has been an area of' extensive research in recent years, focusing mostly on the kinematic motions of single robots in static environments. This problem, known as the *Piano Movers' Problem*, was shown to be solvable in polynomial time [1 1]. A practical geometric representation of the static problem was developed by introducing the concept of *Configuration Space Obstacles* [9]. A few planning methods were developed based on this representation, of which the two notable ones are the *visibility graph [8]*, and the *roadmap* [1].

A practical implementation of an approach similar to the *roadmap* was developed in [13] for two degree-of-freedom manipulators using analytical representations of the configuration space obstacles. in [2], the col]ision-free trajectory is found in the *position-time* space, which is a time extension of the configuration space. in [7], the planning problem is decomposed into two phases: first, a path is selected to avoid the static obstacles; then, the velocity profile along that path is selected to avoid the moving obstacles. In [4], a *collision front* is used to represent the locus of the collision points between the robot and the moving obstacle. An extension of the obstacle avoidance problem is the avoidance of obstacles in minimum time. This is essentially an optimal control problem, which requires the consideration of robot dynamics and actuator constraints [12], [6]. Planning in dynamic environments has also been addressed by the theory of differential games which deals with the computation of trajectorics in state space, under dynamic constraints [5].

In this paper wc focus on using heuristics that are based on the physical behavior of the robot and the environment,. This approach utilizes the concept of the **Velocity Obstacle** (VO) **[3]** that maps the obstacles to the robot's velocity space. The advantages of the velocity obstacle are multi-fold: 1. it facilitates an efficient geometric representation of all possible maneuvers that would avoid the moving obstacle, 2. any number of moving obstacles can be avoided by considering the union of their VO's, **3.** it unifies the consideration of moving as well as stationary obstacles, 4. it allows for the simple consideration of dynamic constraints, and 5. at any given time, it reduces the motion planning problem to a static problem.

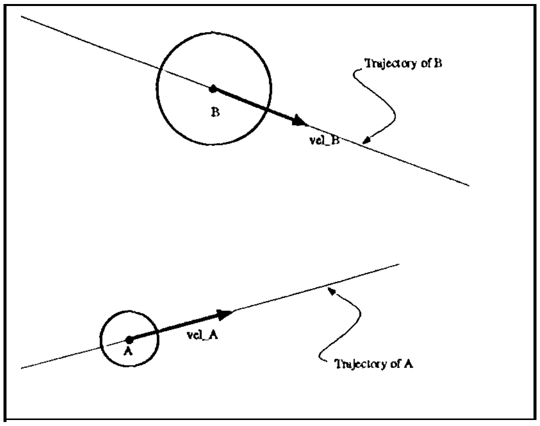

Figure 1: Example of Dynamic l'environment

Since, by definition, the time evolution of the dynamic environment is unknown, the approach to the motion planning problem is naturally heuristic. The geometric representation of all possible avoidance maneuvers permits the efficient selection of those maneuvers that in addition to avoiding collisions, may help accomplish lower priority goals. For example, selecting robot speeds to point toward the final point would guarantee that it reaches its destination. Similarly, selecting the highest feasible speeds would generally reducc motion time. q'bus, heuristics based on this representation ensure that robot motions arc not only safe, but also dynamically feasible and near-time optimal.

In this paper we first review the concept of velocity obstacles and then present two heuristics, the first consisting of selecting the maximum velocity directed to the target, and the second consisting of selecting the maximum feasible velocity within a cone centered at the target. Wc demonstrate the usc of these heuristics to the problem of intelligent vehicles moving from the fast lane to the exit ramp on a smart highway. These solutions are then compared to the ones found by a global search over a tree that represents the time evolution of discrete points in the safe velocity set.

2 Velocity Obstacles and Safe Velocity Sets

The approach proposed for trajectory planning partitions the robot velocity space into two complementary sets. The first one is the velocity *obstacle* consisting of all the velocities causing a collision between the robot and the obstacles. The second one is the *safe velocity set* consisting of the velocities that avoid the collisions and also satisfy dynamic constraints. In this section wc briefly summarize the main features and construction method of these two sets.

We consider a *deterministic* environment, in which obstacles move at specified known speeds. The objective of the planner is to compute a trajectory (path and speed) for the moving robot that accomplishes the following goals: avoid static and moving obstacles, reach a desired destination, and minimize motion time to the destination.

To account for possible uncertainties that may prevent the robot from accomplishing all the above goals [1 O], wc choose heuristics that, first ensure the robot survival in the dynamic environment, and only then attempt to accomplish the remaining goals. Key to this approach is thus the computation of avoidance maneuvers, which arc based on the velocity *obstacle*.

The velocity obstacle provides a graphic representation of the velocity constraints imposed by stationary and moving obstacle. The velocity obstacle extends the concept of the *configuration space obstacles* to the velocity space of dynamic environments.

To introduce the **velocity obstacle, wc** consider two disks, **A** and **B**, moving with arbitrary constant speeds on straight line trajectories, as shown in Figure 1. Disk **A is** the robot, and disk **B** is the obstacle. The four dimensional state space of **A** and **B** is visualized in a two dimensional space, by attaching the velocity vector to their centers, as shown in Figure 1. To determine whether there exists a potential collision between the object and the obstacle, wc first represent the *configuration space obstacle* of **B** by reducing **A** to the point A and growing **B** by the radius of **A** to $\hat{\mathbf{B}}$. Then, wc consider $\hat{\mathbf{B}}$ stationary and $\hat{\mathbf{A}}$ moving at the relative velocity $\mathbf{v}_{A,B} = \mathbf{v}_A — \mathbf{v}_B$. Thc motion of $\hat{\mathbf{A}}$ occurs on the **relativc trajcctory** $trj_{A,B}$ defined as:

$$trj_{A,B} = \{(\mathbf{x}(t), \dot{\mathbf{x}}(t)) \mid \mathbf{x}(t_0) = \mathbf{x}_0, \dot{\mathbf{x}}(t_0) = \mathbf{v}_{A,B}\} \quad (1)$$

A collision will occur between $\hat{\mathbf{A}}$ and $\mathbf{B}$ if thc relativc trajcctory $trj_{A,B}$ intersects $\hat{\mathbf{B}}$ and if both kccp thc same vclocitics. By using $\mathbf{v}_{A,B}$ as a parameter, wc can define thc set of colliding trajectories forming the *Relativc Collision Conc* $\mathbf{CC}_{A,B}$:

$$\mathbf{CC}_{A,B} = \{trj_{A,B} \mid trj_{A,B} \cap \hat{\mathbf{B}} \neq \emptyset\} \qquad (2)$$

Thc cone is the planar sector with apex in $\hat{\mathbf{A}}$, bounded by thc two tangents from $\hat{\mathbf{A}}$ to Î3. Any relativc velocity of **A**, $\mathbf{v}_{A,B}$, remaining within $\mathbf{CC}_{A,B}$, is guaranteed, over time, to cause a collision between **A** and **B**.

Thc relative collision cone defines *indirectly* thc set of colliding absolute velocities, $\mathbf{v}_A$. This set can be spccificd by defining the *A bsolute Collision 12'one, CC_A, which is the* set of absolute velocities, $\mathbf{v}_A$, causing a collision between **A** and **B**. Thc absolute collision cone is obtained by translating the relative collision cone by the vector $\mathbf{v}_B$ [3]. Thus **A** will collide with **B** if the tip of $\mathbf{v}_A$ is inside the cone CC_A. Figure 2 shows the absolute collision cone for disks **A** and **B**. For thc case shown, **A** will collide with **B** since the tip of vector $\mathbf{v}_A$ is inside the velocity obstacle CC_A. Note that the *absolute velocity* cone of a stationary obstacle is identical to its *relativc velocity conc.*

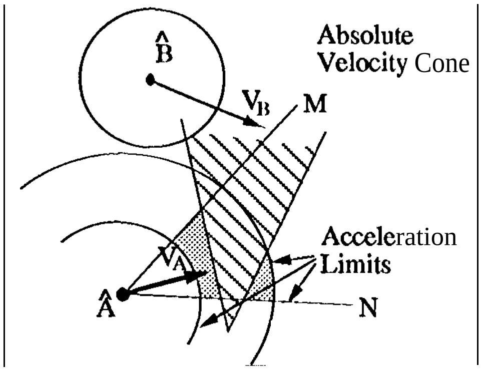

Figure 2: Construction of the velocity Obstacle VO_B

Acceleration constraints, computed from robot dynamics, can bc imposed easily by limiting thc magnitude and direction changes of $\mathbf{v}_A$, as shown in Figure 2. Thc limits on the change in magnitude of $\mathbf{v}_A$ arc represented by circles centered at $\hat{\mathbf{A}}$. Thc direction change limits are represented by the lines M and N. Thcsc dynamic constraints bound the **Feasible Velocity** set $\mathbf{FV}_A(t)$ of **A.**

Thc **Velocity Obstacle** $\mathbf{VO}_B$ of **A due** to **B is** defined as the intersection of the absolutc collision cone $CC_A(t_0)$ with thc feasible velocity set. Thc sct difference bctwccn $\mathbf{VO}_B$ and **I? VA, is the Safe Vclocity set, $\mathbf{SV}_A$**, shown as the grcy areas in Figurc 2. This set consists of all the velocities satisfying both kincmatic and dynamic constraints,

Each velocity $\mathbf{v}^*_A$ whose tip is inside the safe velocity sct $\mathbf{SV}_A$ defines a collision avoiding maneuver at time t_0:

$$(\mathbf{B}(t) \cap \mathbf{A}(t \neq \emptyset) \quad \text{if} \quad (\mathbf{v}^*_A(t_0) \in SV_A(t_0)). \quad (3)$$

Thc avoidance ma cuvcr is guaranteed to avoid the obstacle **B** if **B** maintains its current direction and speed. Selecting the velocity $\mathbf{v}^*_A$ to be on the boundaries of the velocity obstacle $\mathbf{VO}_B(t_0)$ would result in A grazing **B**. Selecting $\mathbf{v}^*_A$ away from these boundaries would ensure some safety margin bctwccn **A** and **B**. Thcsc margins may account for uncertainties about the exact size of II, or its exact velocity.

To avoid several moving obstacles, $\mathbf{v}^*_A$ must be outside the union of their velocity obstacles, forming a **Multiple Velocity Obstacle (MVO)**. **In this** case, thc safe velocity set consists of all thc velocity vcctors whose tip is inside the set diffcrcncc of $\mathbf{F\ V}_A$ and **MVO.**

By observing the position of thc safe velocity set with respect to thc absolutc collision conc, it is possiblc to dctcrminc how disk **A** would avoid disk **B**. We define the front side of disk **B** as the sc~ni-circle in the direction of 13's motion, and its rear side as thc other semi-circle. We define a **front** avoidance mancuvcr when **A** passes in front of **B** and a **rear** avoidance when **A** passes behind **B**. Front and rear avoidance maneuvers arc bounded by those maneuvers grazing **B**, respectively, on its front side and on its rear side. Thc grazing maneuvers corrcspond to the safe vclocities whose tip coinsidcs with on a side of the absolute cone of **B**. If the tip of thc velocity vector is located in the vicinity of the front or of thc rear side of the absolute velocity cone, then thc corresponding maneuver will be, respectively, a front or a rear avoidance mancuvcr. For example, vclocity $\mathbf{v}_A$, shown in Figurc 2, corresponds to a rear avoidance maneuver, since its tip is located near the side of the absolutc velocity conc corresponding to the rear side of disk **B**.

3 Heuristic Planning

We have used the safe velocity sets to develop heuristics for selecting the avoidance velocities that guarantee the survival of the moving robot in the dynamic environment. Since not all velocities within a safe set point to the direction of the goal, the selection of an avoidance velocity reaching the goal must be further refined. We proposed two methods: the first selects the highest safe velocity along the line to the goal, as shown in Figure 3-a, and the second selects the maximum safe velocity within some specified angle from the line to the goal, as shown in Figure 3-b. We will denote the first strategy TG (to goal) and the second one MV (maximum velocity).

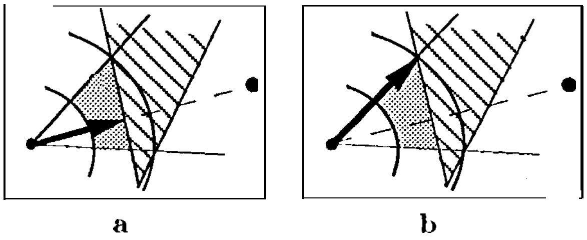

Figure 3: a: TG strategy. b: MV strategy

This approach generates one-step maneuvers that guarantee the survival of the robot if the obstacles maintain their current velocities. It may be necessary to recompute the maneuver in order to reach a specific target or to react to a change in the velocities of the moving obstacles.

The trajectory resulting from these heuristics are conservative, since every trajectory segment is itself a safe trajectory over an infinite time horizon. This is also a drawback of this approach, since it excludes feasible trajectories whose velocities are outside the safe Sets .

An alternative to heuristic planning is to perform a global search for the fastest avoidance maneuvers over all trajectories reaching the goal. To allow a globs] search, the safe sets were discretized at specified time intervals, as shown in Figure 4. This Figure shows a node at time t_0 and its corresponding discretized safe velocity set. The node is expanded by selecting, for example, the two velocities shown, resulting in two different positions and safe velocity sets at time $t_1 = t_0 + \Delta t$. In each of these sets, one velocity is shown as a possible expansion of the nodes to the next time interval. Repeating this expansion for each velocity in every discretized safe set results in a tree that expands at con-

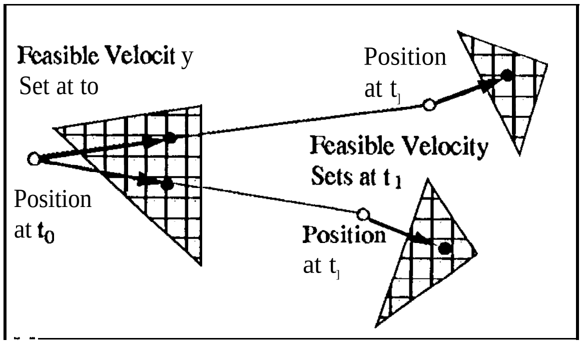

Figure 4: Tree representation for the global search

stant time intervals. A search over this tree allows to select the fastest trajectory reaching the goal,

4 Examples

We applied the proposed strategies to plan the motions of an intelligent vehicle moving from the fast lane to the exit ramp, while avoiding vehicles 2 and 3 in the slower lanes, as shown in Figure 5. Also shown in Figure 5 is the trajectory computed with the TG heuristics. Along this trajectory, vehicle 1 slows down and lets vehicle 3 pass, and then speeds up towards the exit, behind vehicle 2. The total motion time for this trajectory is $5.8s$.

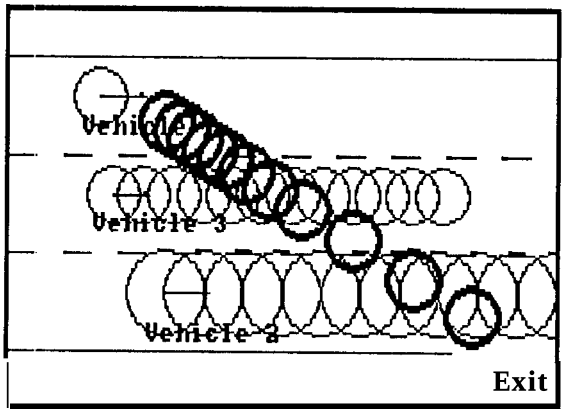

Figure 5: Trajectory with the TG strategy (Δt = .25s)

The solution computed with the MV heuristics is shown in Figure 6. Along this trajectory, vehicle 1 speeds up and passes in front of both vehicles 3 and 2. The total motion time along this trajectory is $3.9s$.

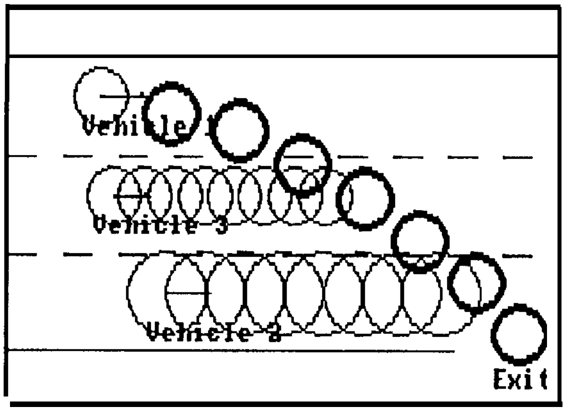

Figure 6: Trajectory with MV strategy ($\Delta t = 1s$)

In the case of Figure 6, the MV heuristics was successful to reach the goal, but this is not guaranteed, 'J'o improve the chances of reaching the goal, it may be useful to combine the TG and MV heuristics by selecting the TG velocity if it exceeds some minimum limit and the M V velocity otherwise.

The trajectory computed using both MV and TG strategies is shown in Figure 7. Along this trajectory, vehicle 1 first speeds up to pass vehicle 3, then slows down to let vehicle 2 pass on, and then speeds up again towards the goal. The motion time for this trajectory is 5.6s. This trajectory was computed by using the MV heuristics for O $\leq t \leq$ 2 and the TG heuristics afterwards.

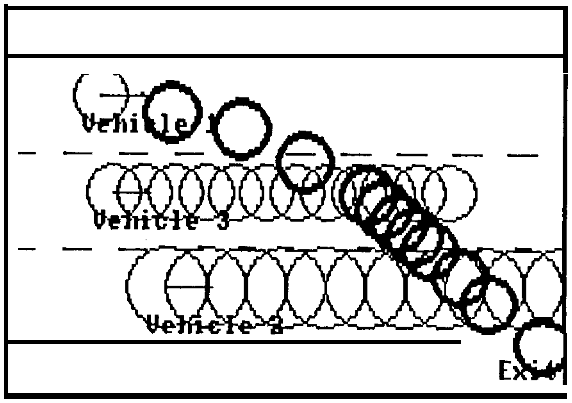

Figure 7: Trajectory with TG,MV strategies ($\Delta t = 1s$)

Figure 8 shows the absolute cones of vehicle 2 and 3 at time $t =$ 2s of' the trajectory shown in Figure 7. Also shown as a dashed line, is the straight line from vehicle 1 to the goal. Polygons A and 13, bounded by the darker lines, represent the safe velocity sets of vehicle 1 at that specific time. In this case, the line to the goal intersects the safe velocity set A, This implies that, at this point, we could usc either the TG or the MV heuristic.

Figure	Heuristics	$\Delta t(s)$	total time(s)
5	TG	.25	5.8
6	MV	1	3.9
7	MV,TG	1	5.6
9	none	1	5.4

Table 1: Summary of trajectory parameters

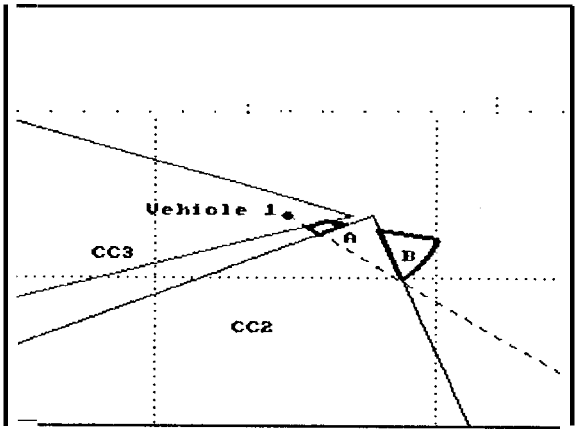

Figure 8: Safe velocity sets

The TG heuristic would choose the velocity along that line, and compute the trajectory shown in Figure 7. The MV heuristics would select a larger velocity in set B, pointing at some angle from the straight, line to the goal, and compute the trajectory shown in Figure 6. In this case, the MV heuristics would result in a faster trajectory.

By observing the safe velocity set shown in Figure 8, it is possible to determine how each maneuver would avoid vehicles 2 and 3. The safe set, A, is bound by the front side of the absolute cone of vehicle 3 (CC3), and by the rear side of the absolute cone of vehicle 2 (CC2). 'J'bus, the velocities in set A correspond to the front avoidance of vehicle 3 and to the rear avoidance of vehicle 2. The velocities in set B correspond to the front avoidance of both vehicles 2 and 3. Recognizing front aud rear maneuvers allows the planner to choose conservative and risky avoidance maneuvers. in this example, vehicle 2 may represent a big truck. It might be therefore safer to avoid vehicle 2 by passing behind it, as shown in Figure 7, rather than pass in the front, as shown in Figure 6.

The trajectory in Figure 9 was computed try a global search, expanded to 6s with $\Delta t = 1s$. The motion time for this trajectory is 5.4s. For this example, the motion time was not better than the heuristic trajectories, probably clue to the low resolution used in the discretization of the safe velocity sets. A summary of the parameters for the various trajectories is shown in 'l'able 1.

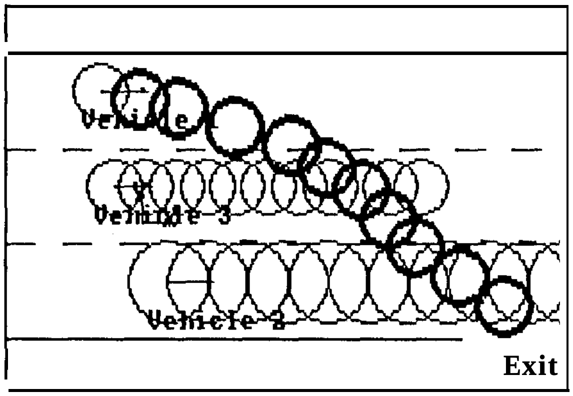

Figure 9: Trajectory computed with global search

5 Conclusion

We developed two heuristic strategies for trajectory planning in a dynamic environment based on the concept of Velocity Obstacle. The first selects the highest avoidance velocity in the direction to the goal (TG) and the second selects the highest, velocity within a prescribed cone towards the goal (M V). Both strategies attempt to satisfy a hierarchy of goals, whose first objective is to guarantee the survival of the robot then to reach a target point and, lastly, to minimize motion time. Both strategies ensure the robot's safety, but they do not, guarantee the satisfaction of the higher level objectives. These heuristics were demonstrated for a vehicle moving from the fast lane to the exit ramp while avoiding two vehicles moving in the other lanes. in the case shown, the MV heuristics resulted in faster and riskier trajectory than the trajectory computed by the TG heuristics. '] 'hese strategies have the potential of providing useful and efficient planning heuristics for the automatic control of intelligent vehicles moving on smart highways.

Acknowledgment

The research described in this paper has been partially carried out at the Jet]'repulsion Laboratory, California Institute of Technology, under contract, with the National Aeronautic and Space Administration.

References

[1] J. Canny. *The Complexity of Robot Motion Planning,* MIT Press, Boston, MA, 1988.

[2] M, Erdmann and T. Lozano-Perez. On multiple moving objects. in *IEEE International Conference on Robotics and Automation,* pages 1419-1424, San Francisco, CA, April 7-10 1986.

[3] P. Fiorini and Z. Shiller. Motion planning in dynamic environments using the relative vleocity paradigm. In *IEEE International Conference* on *Robotics and Automation,* Atlanta, GA, May 7-101993.

[4] K. Fujimura and H. Samet. Time-minimal paths among moving obstacles. In *IEEE International Conference* on ***Robotics and Automation,*** pages 1110-1115, Scottsdale, AZ, May 1989.

[5] 1{. Isaacs. *Differential Games.* John Wiley & Sons, 1965.

[6] D.W. Johnson and E.G. Gilbert. Minimum time robot **path planning** in the presence of obstacles. in *24th Conf. on Decision and Control,* Ft.], Lauderdale, FL, December 1985.

[7] K, Kant and S.W. Zucker. Towards efficient trajectory planning: the path-velocity decomposition. *The International Journal of* ***Robotic*** *Research,* 5(3):72- ***89,*** Fall 1986.

[8] J.C. Latombe. ***Robot Motion Planning,*** Kluwer Academic Publisher, Boston, MA, 1991.

[9] 'J'. Lozano-Pérez. Spatial planning: a configuration space approach. *IEEE Transaction on Computers,* C-32(2):108- ***120,*** February 1983.

[10] J.C. Sanborn and J.A. Hendler. A model of reaction for planning in dynamic environments. *International Journal of* ***Artificial*** *Intelligence* in *Engineering,* 3(2), April 1988.

[11] J. Schwartz, M. Sharir, and J. Hopcroft. *Planning, Geometry and Complexity* oj Robot *Motion.* Ablex Publishing Corp., Norwood, New Jersey, 1987.

[12] Z. Shiner and S. Dubowsky. Time optimal path planning for robotic manipulators with obstacles, actuator, gripper and payload, constraints. international *Journal of* Robotics *Research,* December 1989.

[13] Z. Shiller and R.Y. Gwo. Collision-free path planning of articulated manipulators. *A SME* ***Journal of*** *Mechanical* *Design,* December 1993. To appear.